設計自動化　54－3
(1990. 10. 8)

検査容易性による故障の分類と順序回路のテスト生成について

梶原誠司[+]　ｱﾝﾄﾆｵ.ﾙﾋﾞｵ[++]　樹下行三[+]

[+]大阪大学工学部　応用物理学科
[++]イリャスバレアレス大学，スペイン

あらまし　本報告では，順序回路のスタックオープン故障を検出するテスト生成手法を提案する．テストパターンは，ＰＯＤＥＭ法のバックトレースによる入力値割当てと，演繹故障シミュレーションによる前方操作を用いて生成する．テスト生成では，活性化されている故障と故障リストが伝搬されている信号線の検査容易性の尺度により，目標故障の変更を行ない，ＣＯＮＴ法と同様にバックトラックを行なわない手法を導入する．また本報告では，演繹故障シミュレーションの効率化について考察し，順序回路の初期状態問題についても議論する．最後にベンチマーク回路を用いてテスト生成手法を評価する．

Fault classification by testability and test generation for sequential circuits

Seiji Kajihara[†]　Antonio Rubio[††]　Kozo Kinoshita[†]

[†]Osaka University
2-1 Yamada-oka, Suita 565, Japan

[††]University of the Balearic Islands
Cta. Valldemossa Km.7.5
07071, PALMA, Spain

abstract: In this paper, we describe a test generation method to detect stuck-open faults in sequential circuits. Test patterns are generated by using the backtrace of PODEM to assign a value of primary inputs and by using the deductive fault simulation to assign a value of internal lines. In this method, we switch target fault by the testability measure of sensitized faults and signal lines where fault lists have been propagating. As a consequence, this method performs no backtracking like as in CONT algorithm. We also consider a method to increase the efficiency of deductive fault simulation and discuss initial state information problems for stuck-open faults in a sequential circuit. Finally we evaluate the proposed method for benchmark circuits.

1．はじめに

順序回路のテスト生成は，スキャンパス設計等の検査容易化技術を用いることにより，組合せ回路の故障検査と同様に扱うことができる．しかし，スキャンパス設計等を用いる場合，チップ面積の増加，回路性能の低下等の問題が生じるため，部分的にスキャンパス設計を適用する手法[1-3]，順序回路から直接テストパターンを生成する手法[4-9]が提案されている．順序回路から直接テストパターンを生成する手法の多くは，回路内のフィードバックループを切断し，各時間の回路動作を直列に接続することで，組合せ回路のテスト生成手続き［10,11］を用いている．信号値を従来の5値から9値にする手法[4]，バックトラックを少なくするための解析を行なう手法［5,6］は，Dアルゴリズム［10］の拡張である．PODEM[11]に基づく手法には，知識ベースを取り入れたHITEST［7］，PODEMに9値法を取り入れたFASTEST［8］がある．これらの手法は，回路の時間展開のため必要な記憶容量が増大すること，テスト生成手続きが複雑になることが問題になる．テスト生成を故障シミュレーションに基づいて行なう手法も提案されている［9］．故障シミュレーションを用いることで，遅延情報等ゲートの時間動作も扱いが容易になる．

これらのテスト生成は，いずれも縮退故障を対象にテストパターン生成を行なっている．本稿では，CMOS順序回路のスタックオープン故障に対するテストパターン生成法を提案する．スタックオープン故障は，組合せ素子に記憶素子の動作を起こさせるため，その検出には，連続する2つのパターンが必要であり，故障の正確なモデル化にはスイッチレベルの回路情報が求められる[12-14]．これまでの報告で，スタックオープン故障に対するテスト生成をCMOSゲートの入出力におけるテストパターンをスイッチレベルで生成し，回路全体のテストパターンはゲートレベルで生成する手法を提案してきた[15]．本稿では，その手法を順序回路に適用する．ゲートレベル回路で行なうテスト生成は，CONT法[16]に基づいて行なう．CONT法は，PODEM法を改良したテスト生成手法であるが，前方操作において同時故障シミュレーションと同様の方法で故障リストを作成し，目標故障の入力値の割当てに失敗したときは，生成されている故障リストを用いて目標故障を変更することでバックトラックを行なわないことに特徴がある．本手法では，目標故障の変更を入力値の割当てに失敗したときに行なうのではなく，故障リストが更新される毎に検査容易性の尺度を参照して行なう．

次の2節では，テスト生成の手法について説明し，故障リストを計算する故障シミュレーションの効率化手法についても提案する．3節では，スタックオープン故障と順序回路における検出可能性について議論する．また，検査容易性の尺度とそのテスト生成への活用について述べる．4節では，ベンチマーク回路に対する実験結果を報告し，最後の5節でまとめを行なう．

2．テスト生成手法

2．1　概要

本稿では，対象回路として，記憶素子としてDフリップフロップを含む単一クロックの同期式順序回路を仮定する．Dフリップフロップはゲートレベル回路においては，1つのCMOSゲートとして扱う．提案する手法は，バックトレースによる外部入力の値の決定と，その外部入力からのイベント駆動式故障シミュレーションによる故障リストの伝搬の2つの基本操作の繰り返しにより，テストパターンを生成する．

2．1．1　信号値

正常回路のシミュレーションに用いる信号値は，0，1，U，X，U0，U1の6種類である[17]．信号値Uは，0または1に確定しているが，どちらにも特定できない信号値で，記憶素子の出力における未知の初期状態等を表わす．テスト生成は，初期状態としてのフリップフロップの出力値Uに対する含意操作から始まる．信号値Xは，未定義を表わす．信号値U0とU1は，信号値Xに制限を加えた信号値で，信号値U0はUまたは0にしかならない未定義値を，信号値U1はUまたは1にしかならない未定義値を表わす．信号値U0とU1を用いる目的は，信号値Uの伝搬を避けるようなテストパターンを生成する手助けにすること，及び，バックトレースによる外部入力値の決定における無駄を少なくすることにある．

2．1．2　初期状態問題

順序回路のテスト生成では，フリップフロップ等の記憶素子の初期状態が未知のままテスト生成を行なうことは難しいため，一般には，仮定故障に対するテスト生成の前に，回路の内部状態を既知にするためのテストパターンを生成する．このとき，次に示す初期状態問題が存在する[8]．

［初期状態問題］もし既知になる状態が故障の無い回路のシミュレーションで決められるか，または，テスト生成による初期化が故障の無い回路を仮定して行なわれるなら，結果として得られた状態は故障の存在下では正しくないかも知れない．この（正しくないかもしれない）状態を正常回路と故障回路の両方に仮定して生成されたすべてのテストは，無効になると考えられなければならない．

スタックオープン故障を考える場合，最初に印加されるパターンに対して，ゲート出力をハイインピーダンス状態にする故障が存在すれば，その時のゲート出力値は特定できないため，初期状態問題はより複雑になる．本手法では，回路の内部状態を既知にするパターンについ

ても故障シミュレーションを行なうこと，また，故障シミュレーションでは検出可能な故障のリストだけでなく，故障時に信号線の値をUにする故障のリストを，通常の故障リストと同様に生成し伝搬することで初期状態問題を解決する．

2．1．3　目標信号線の選択

外部入力値は，PODEM法で用いられるバックトレースにより求めるが，その始点にあたる目標信号線と信号値は，

1）内部状態を既知にするための信号線と信号値

2）生成されている故障リストを伝搬するための信号線と信号値

を選択する．回路全体の90%以上の内部状態が決定するまでは，1）を優先的に選択する．この場合フリップフロップの入力に論理値0または1が設定されるようにする．その後，故障リストが生成されているときは2）を選ぶ．故障リストを持つ信号線がないときは信号値の値が未定である外部入力にランダムに値を設定する．一般には，複数の信号線が故障リストを持つと考えられるが，それらの中から検査容易性の尺度を参照に目標信号線を決定する．これについては3．3節で説明する．

2．1．4　含意操作と故障リスト生成

バックトレースにより決定した外部入力値による内部信号の変化を求める操作では，同時に演繹故障シミュレーションと同様の操作を行い，その信号線で検出可能な故障の集合を求め，故障リストに記述する．故障リストが生成された後は，特定の対象故障を持たないため，Dアルゴリズム等で用いられる信号値D，－Dは使わない．また，目標信号線は，含意操作と故障リスト生成が行なわれる毎に，新しく決定するため，CONTアルゴリズムと同様に外部入力におけるバックトラックは行なわない．

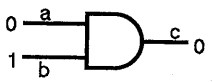
2．2　故障リスト伝搬

故障リストの生成，伝搬による故障シミュレーション手法には，演繹法[18]，同時法[19]等があるが，本手法では演繹法を用いる．演繹法は，各信号線で検出可能な故障を，入力側から故障集合の集合演算により，演繹的に求める手法である．例として，図1のように，2入力ANDゲートにa＝0，b＝1の入力が印加された場合を考える．対象が縮退故障の場合，信号線a，bで検出可能な故障の集合をそれぞれLa，Lbとする時，信号線cで検出可能な故障集合Lcは

$$Lc = (La \cap \overline{Lb}) \cup \{c/1\}$$

で求められる．ここで，c/αは信号線cのα縮退故障を表わす．

演繹法は，故障集合が大きくなった場合，計算機上での集合演算の計算量が多くなることが欠点の一つと考えられてきた．本稿では，スタックオープン故障を対象とするため，縮退故障の場合に比べ，同じパターンで活性化される故障は少なくなる．また，検出済みの故障については考慮しない．これらのことから，各信号線で検出可能な故障集合の大きさは，比較的小さくなると考えられる．



$Lc=(La\cap\overline{Lb})\cup\{c/1\}$

Fig.1 Deductive Fault Simulaion.

次に，回路情報に基づいた演繹法の集合演算の簡単化について述べる．

2．2．1　非再収れんゲート

演繹法において検出可能な故障集合を求める時，多くの計算量を必要とする演算は，（La∩Lb）のような同じ故障が複数の入力信号線に伝搬しているか否かを調べるための演算である．もし，シミュレーションを行なうゲートの異なる入力に，同じ故障が伝搬しないことがわかっていれば，集合演算の簡略化が可能である．例えば，図1のANDゲートの信号線a，bに同じ故障が伝搬しないならば，故障集合Lcは

Lc＝La∪｛c／1｝

で求められる．

ゲートの異なる入力に同じ故障が伝搬しないための十分条件は，そのゲートがどの信号線の再収れんポイントにもなっていないことである．ある信号線の再収れんポイントになっているゲートを再収れんゲート，そうでないゲートを非再収れんゲートとすると，非再収れんゲートである2入力ANDゲートの故障集合Lcを求める演算は，各入力組合せに対し表1のように簡略化できる．

a	b	Lc
0	0	{c/1}
0	1	La∪{c/1}
1	0	Lb∪{c/1}
1	1	La∪Lb∪{c/0}

Table1 List Propagation.

組合せ回路の再収れんゲートは，各分岐点からの有向パス（directed path）をすべて検索することにより，調べることができる[20]．順序回路では，各入力から同様に有向パスを検索することで，容易に得ることができる．そこで，本手法では，回路内のすべてのゲートについて，前もって再収れんゲートを調べることにより，非再収れ

んゲートに異なる集合演算を適用し故障シミュレーションの効率化を試みる．

2．2．2　伝搬フラグ

ここでは，故障シミュレーションにおける，無駄なリスト伝搬を行なわないための伝搬フラグについて説明する．故障リストの生成と伝搬は，バックトレースにより決められた外部入力の値について，イベント駆動方式で行なう．例えば，図2回路で，a＝1，c＝1，d＝0が既に定まっていると仮定する．このとき，信号線hで検出できる故障集合Lhは，信号値gの値が決定するまで求められない．しかしながら，この場合，ゲートG3は非再収れんゲートであり，またd＝0であるため，Lgの故障集合はLhに影響を及ぼさない．つまり，信号値bの決定により，信号線f，gの値が決まったとしても，故障リストLf，Lgを計算する必要は無い．これは，d＝0が決められたときに，判断できることである．そこで，本手法では，無駄な故障リストの伝搬を避けるために各信号線に伝搬フラグを導入する．

伝搬フラグは，GO，STOP，PAUSEの3種類の信号のいずれかを持つ．信号値が未定義の時のフラグはPAUSEであり，その信号線で検出できる故障集合がまだ求められていないことを示す．信号線で検出できる故障集合が求められた時GOになる．フラグがSTOPである信号線は，その信号線で検出可能な故障集合を求める必要が無いことを示す．フラグが，STOPになるための条件を示す．

2入力以上のゲートについて，

1）ゲートの正常時の入力の信号値がUであるとき，または，

2）非再収れんゲートの入力値が，同じゲートの他の入力値にかかわらずそのゲートの出力値を決める値であるとき，または，

3）再収れんゲートの入力値が，同じゲートの他の入力値にかかわらずそのゲートの出力値を決める値であり，かつ，その入力信号線で検出可能な故障が無いとき，

以上の3通りの場合，同じゲートの他の入力信号線のフラグはSTOPになる．更に，

4）ゲート出力のフラグがSTOPのとき，そのゲートのすべての入力信号線のフラグは，STOPになる．

5）分岐点において，すべての枝のフラグがSTOPのとき，幹のフラグは，STOPになる．

伝搬フラグがGOからSTOPに変化する場合，その信号線の故障リストは，外部出力に伝搬される可能性が無くなるため，削除できる．

図2の回路では，d＝0の含意操作によりh＝0が決まったとき，条件2）より，信号線gの伝搬フラグはSTOPになる．これに伴い，条件4）より信号線f，cのフラグもSTOPになり，信号線cに故障リストLcがある場合，Lcは削除される．

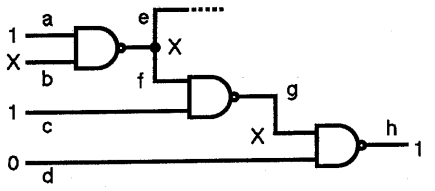


Fig.2 Circuit Example.

2．3　アルゴリズム

テスト生成の基本アルゴリズムを図3に示す．まず，初期状態としてのフリップフロップの出力値Uに対する含意操作を行なう．次に，信号値設定の目標になる信号線を選択する．回路の内部状態が決まるまでは，フリップフロップの入力が目標信号線になり，その後は故障リストを伝搬させる信号線が目標信号線になる．この目標信号線に対し，バックトレースにより外部入力値を決定し，その値による含意操作と故障リストの生成，伝搬の操作を行なう．このとき，伝搬フラグの計算も行なう．テストパターンの終了条件は，本稿では，すべての故障が検出されるか，5パターン連続して新たな故障が検出されない場合とした．

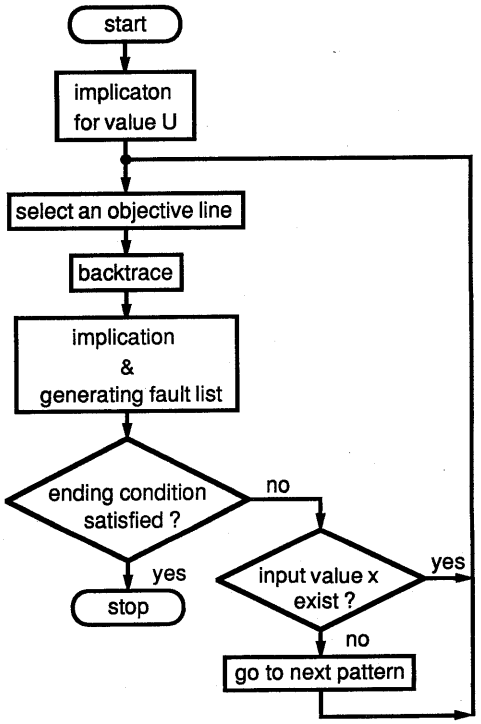


Fig.3 Flowchart of test generation.

3．　故障の分類

3．1　スタックオープン故障

3．1．1　2パターンテスト

組合せ回路におけるＣＭＯＳトランジスタの単一スタックオープン故障の検出には，連続する２つのパターン＜Ｔ１，Ｔ２＞が必要なことが知られている[13,14]．表２には，図４のＣＭＯＳ２入力ＮＡＮＤゲートのスタックオープン故障の検出に必要な２パターン＜Ｔ１，Ｔ２＞を示す．Ｔ１は，故障を仮定するゲートの出力値として，正常時にＴ２が印加されたときの出力値と反対の論理値を持つように生成されるパターンである．Ｔ２は，仮定した故障によりゲート出力がハイインピーダンス状態になるように生成される．またＴ２では，ゲート出力に現われる故障の影響を外部出力に伝搬することも要求される．本稿では，Ｔ１に相当するパターンを初期化パターン，Ｔ２に相当するパターンを検出パターンと表わすことにする．

また，故障の正確なモデル化にはスイッチレベルの回路情報が求められる．そこで本手法では，ＣＭＯＳゲートの入出力におけるテストパターンをスイッチレベルで生成し，回路全体のテストパターンはゲートレベルで生成する手法を適用する[15]．

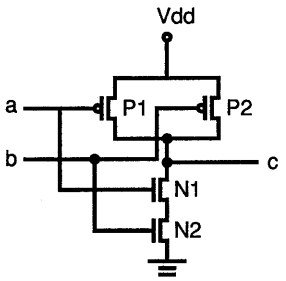


Fig.4 CMOS NAND Gate.

fault	T1 (a,b)	T2 (a,b)
P1	(1,1)	(0,1)
P2	(1,1)	(1,0)
N1, N2	(0,x) (x,0)	(1,1)

Table 2　Test Sequences.

3．1．2　冗長故障と検査不能故障

スタックオープン故障は，初期化パターンと検出パターンのいずれかが生成できない場合には検出できない．しかしながら，テストにより検出されない故障のすべてが回路出力に影響をしない故障であるとは限らず，回路出力に影響する可能性のある故障もある．例えば，図５のＮＡＮＤゲートＧ１で，そのゲート入力にＴ１＝（１，１），Ｔ２＝（０，１）が必要なスタックオープン故障を仮定する．Ｔ１の印加により，Ｔ２の入力ｂの値は必ず０になるため，この故障は検査できない．しかしながら，直前のパターンがＴ１でない場合，Ｔ２は生成可能である．Ｔ２が連続して印加され，ゲートの出力が長い時間ハイインピーダンス状態に保たれた場合，ゲートの出力値は論理的に不安定になる．このような故障はテストによる故障の検出は保障されないが，故障の影響が回路に現れる可能性を持っている．

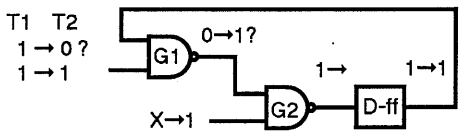


Fig.5 Circuit Example.

本稿では，テストにより検出されない故障のうち，回路出力に影響をしない故障を冗長故障，回路出力に影響する可能性のある故障を検査不能故障として区別する．

テストにより検出されない故障のうち，冗長故障は，初期化パターンの有無に関係なく，検出パターンとなりうる入力パターンが存在しない場合である．このとき，ゲート出力をハイインピーダンスにする入力パターンは存在しないため，故障の影響が回路内に現われることはない．

検査不能故障には，次の２通りの場合が考えられる．

［場合１］　検出パターンは存在するが，初期化パターンは存在しない場合．

［場合２］　単独には検出パターンが存在するが，初期化パターンに連続する検出パターンは存在しない場合．

図５の検査不能故障は，［場合２］に含まれる．［場合２］の検査不能故障は，回路に記憶素子とループ構造を有する順序回路に固有の故障である．

冗長故障と検査不能故障は，テスト生成の効率を悪くするとともに故障検出率を低下させる要因になる．本手法においては，テストパターン生成の前に，各スタックオープン故障について，初期化パターンと検出パターンに対する含意操作を行い，比較的容易に判断できる冗長故障と検査不能故障を仮定故障から取り除く操作を行う．

3．1．3　記憶素子の故障

記憶素子内のスタックオープン故障については，クロック信号線に接続されたトランジスタの動作を考慮する必要から，通常の１パターンを，クロックがオンの時とオフの時の２つに分解して解析する[21,22]．例えば，図６に示されるＤラッチのトランジスタＰ１とＰ２のスタックオープン故障の検出には，表３のテストパターンが必要である．このとき，クロックがオフからオンに変化するときは，入力Ｄは変化しないことが要求される．一般に，ラッチ，フリップフロップ等の記憶素子内部には，

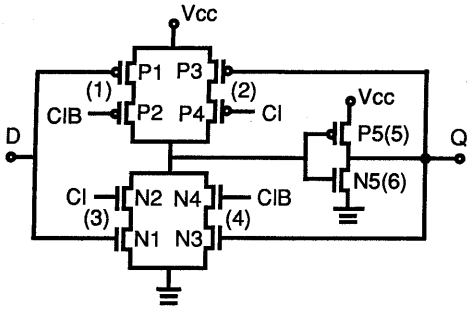


Fig.6 D-Latch circuit.

	Test		fault-free		faulty	
	Cl	D	Q2	Q1	Q2	Q1
T1	1	1	0	1	0	1
T2	0	0	0	1	0	1
T3	1	0	1	0	0	1
T4	0	X	1	0	0	1

Table 3 Test sequence for stuck-open fault of transistor P1 or P2.

フィードバックループが存在するため，冗長故障や検査不能故障がある．図６のＤラッチでは，パス（２）と（４）上のトランジスタの故障は上記の［場合２］に相当する検査不能故障である．これらの故障は，付加トランジスタの挿入により検査することができる[21, 22]．

本稿では，回路内に含まれる記憶素子として，図６のＤラッチを直列に接続することにより得られるマスタースレイブ型のＤフリップフロップを仮定する．この形状のフリップフロップは，１６の仮定故障に対して４個の検査不能故障がある．これらの故障は，マスター側とスレイブ側の各ラッチに，それぞれ４個のトランジスタと１本の入力を付加することにより，検出可能になる[21]．

３．２　検査容易性

順序回路のテスト生成を困難にする要因には，故障の設定操作に複数のパターンが必要なこと，また，故障の影響を外部出力に伝搬するためにも複数のパターンが必要なことがある．しかしながら，検出のための複雑さ（テスタビリティ）は各故障により異なり，故障の設定操作が容易な故障，外部出力への伝搬が容易な故障等に分けることができる．本手法では，各信号線について，信号値の設定に複数のパターンを要するか，その信号線の値の伝搬に複数のパターンが必要かを調べ，その情報をテスト生成に活用することを試みる．

まず，可制御性について，信号線を２つの集合に分類する．

集合Ｓｉ：どのフリップフロップの出力からも有向パスが存在しない信号線の集合．

集合Ｓｓ：少なくとも１つのフリップフロップの出力からの有向パスが存在する信号線の集合．

すべての信号線の集合をＵで表わすとき，Ｓｓ＝Ｕ－Ｓｉである．Ｓｉに含まれる信号線は，入力側へのパスのすべてがフリップフロップを通過する事なく外部入力に接続されているため，その信号値の制御は，１つのパターンで行なうことができる．

次に，可観測性について，信号線を３つの集合に分類する．

集合Ｋｏ：どのフリップフロップの入力へも有向パスが到達しない信号線の集合．

集合Ｋｓ：すべての有向パスがフリップフロップの入力へ到達する信号線の集合．

集合Ｋｍ：外部出力とフリップフロップの入力の両方へ有向パスが到達する信号線の集合．Ｋｓ＝Ｕ－Ｋｏ－Ｋｓである．

集合Ｋｏに含まれる信号線上の故障は，外部出力への故障の伝搬を１パターンで行なうことができる．集合Ｋｓに含まれる信号線上の故障は，外部出力への故障の伝搬に２パターン以上必要である．

３．３　目標信号線の決定

回路の内部状態を既知にするパターンを生成した後，バックトレースの始点となる目標信号線は，生成されている故障リストを伝搬するために必要な信号値を設定するように選択される．一般に故障リストを持つ信号線は複数存在するが，各故障リストを信号線の可観測性とリストに含まれる故障の可制御性，可観測性によりクラス分けを行い，伝搬の優先順位を決める．

各ゲートの出力信号線の可観測性と可制御性から，そのゲート内で起こるスタックオープン故障の検査容易性のクラスを表４のように定義する．次に故障リストの存在する信号線の可観測性とリストに含まれる故障の検査容易性から，各故障リストを表５のように定める．例えば，クラスＡは，集合Ｏｏに含まれる信号線で検出できる故障のリストで，リストにはクラスＦａの故障が含まれており，Ｆａ以外の故障は括弧内のＦｂまたはＦｃの故障であることを示す．故障リストの伝搬のときのバックトレースの始点は，クラスＡ，Ｂ，Ｃ，・・・，Ｉの順に選択する．つまり，外部出力への伝搬が容易な故障リストの伝搬を優先し，また，検出が難しいと考えられる故障の早期検出を目標とすることになる．

４．　実験結果

本手法を用いて，順序回路のベンチマーク回路に対するテストパターン生成の実験を行なった．実験回路は，大きなフィードバックループの存在により，表６に示す

class of fault	controllability of output line	observability of output line
Fa	Ss	Ks
Fb	Ss	Km,Ko
	Si	Ks,Km
Fc	Si	Ko

Table 4 Fault classificaton.

class of list	observability of line	class of included faults
A	Ko	Fa (Fb,Fc)
B	Ko	Fb (Fc)
C	Ko	Fc
D	Km	Fa (Fb,Fc)
E	Km	Fb (Fc)
F	Km	Fc
G	Ks	Fa (Fb,Fc)
H	Ks	Fb (Fc)
I	Ks	Fc

Table 5 List classification.

回路における２入力ゲートの数と再収れんゲートの割合，そして，各信号線の可観測性についての分類の結果を示す．大きなフィードバックループの存在により，２入力以上のゲートの多くは再収れんゲートであった．このため，これらの回路では，非再収れんゲートにおける故障リストを求める演算の簡略化の効果は現れない．しかしながら，フィードバックループを含まない組合せ回路では，非再収れんゲートの割合が増加すると考えられ，演算の簡略化の効果が期待できる．

ベンチマーク回路は，ゲートレベルで記述されているが，本稿では，各論理ゲートが１つのＣＭＯＳゲートを構成するとして，スタックオープン故障を仮定した．同一ゲートに含まれる等価な故障を１つとした場合の，回路内に仮定するスタックオープン故障の数を表７に示す．本手法では，テストパターン生成の前に，簡単な含意操作により判断できる検査不能故障と冗長故障を調べた．どの実験回路に対しても，前処理により検出される冗長故障は無く，検査不能故障は表７に示す通りであった．ここに示す冗長故障数と検査不能故障数は，回路内のすべての冗長故障や検査不能故障の数ではないが，ＳＯＣＲＡＴＥＳ[24]で行なわれているような詳細な解析を行なうことにより，検査不能故障や冗長故障と判断できる故障を増やすことができると思われる．

順序回路のテスト生成では，フィードバックループがあるために，過去に活性化された同じゲートの故障の影響を考慮する必要がある．スタックオープン故障を仮定する場合，初期化パターンと検出パターンのそれぞれについて調べる必要から，直前のパターンにおける各信号線の正常時と故障時の値を保持しなければならず，故障時の出力値の計算も複雑になる．そこで，本稿では，ある故障がその故障を含むゲートに影響を及ぼす場合のゲートの出力値をＵに仮定し，また，初期化パターンにおいては，故障の影響を考慮しないでテストパターンを生成した．

表７のパターン数には，生成したパターンの数と，そのうち回路の内部状態の初期化に要したパターン数を括弧内に示す．検出率は

$$\%\text{coverage} = \frac{\#\text{untestable} + \#\text{detected}}{\#\text{faults}} \times 100$$

の式により算出した．本手法のテスト生成では，故障リストを伝搬するゲートの入力値と伝搬フラグの値がすべて決まるまで，リストが伝搬できないため，それらの値を決定するための外部入力値が，故障リストを外部出力に伝搬するときの妨げになる場合がある．また，通常の故障リストと同時に，故障時に信号値をＵにする故障のリストを伝搬するが，このリストが大きくなり，全体のテスト生成の効率を悪くする要因になっている．より高い検出率を得るためには，これらのテストパターン生成に引続き，経路活性化法に基づくテスト生成を行なう事などが必要である．

name	# of gates	% of reconvergence	% of lines		
			Ks	Km	Ko
S27	8	100	27	65	8
S298	75	100	90	2	8
S344	101	93	82	9	9

Table 6 Benchmark circuits.

５．まとめ

本報告では，順序回路のスタックオープン故障を検出するテスト生成手法について述べた．テストパターンは，バックトレースによる入力値割当てと，演繹故障シミュレーションによる前方操作を用いて生成し，活性化されている故障と故障リストが伝搬されている信号線の検査容易性から，目標故障を変更する手法を提案した．また，演繹故障シミュレーションの効率化手法についても提案し，ベンチマーク回路を用いた実験からテスト生成手法を評価した．今後の課題として，入力値の割当てに多重バックトレース等を導入することで，より効率的にテス

name	# of faults	# of redundant	# of untestable	# of patterns	% of coverage	time
S27	34	0	4	10(1)	76.4	0.07
S298	391	0	11	49(2)	38.6	12.5
S344	459	0	16	56(2)	71.2	6.4

Table 7 Experimental results.

ト生成を行なうことが考えられる．また，高い検出率を得るために，未検出故障については経路活性化法に基づくテスト生成を行なう必要がある．

謝辞

最後に本研究を行なうにあたり，始終討論して頂いた本学樹下研究室助手板崎徳禎博士，ならびに作図等にご協力頂いたＭ１芝温子さんに深く感謝します．

参考文献

(1) V.D.Agrawal, K.-T.Chen, D.D.Johnson,T.Lin, "Designing Circuits with Partial Scan,"IEEE Design &Test,vol.5,No.2,pp.8-15,April 1989.

(2) T.Ogiwara, "Test Generation System for Sequential Circuits," FTCS-18,Poster Session,1988.

(3) 奥村憲三，野田浩明，松本比呂志，"部分スキャン方式の自動テストパターン生成"，第２１回ＦＴＣ研究会資料，July 1989.

(4) P.Muth, "A Nine-Value Circuit Model for Test Generation,"IEEE Trans.Comput.,vol.C-25,pp.630-636 ,June 1976.

(5) R.Marlett,"EBT,An Effective Test Generation Systems for Sequential Circuits," 23rd DAC,pp.250-256,1986.

(6) S.Mallela and S.Wu, "A Sequential Test Generation Systems," ITC,pp.57-61,1985

(7) M.J.Bending,"HITEST,A Knowledge-base Test Generation System," IEEE Design&Test of Comput., vol.1,pp.83-93,May 1984.

(8) T.P.Kelsey, K.K.Saluja,"Fast Test Generation for Sequential Circuits," ICCAD,pp.354-357,1987.

(9) V.D.Agrawal, K.T.Cheng and P.Agrqwal,"A Direct Search Method for Test Generation Using A Concurrent Simulator," IEEE Trans. CAD,vol.8,pp.131-138,Feb. 1989.

(10) J.P.Roth,"Diagnosis of Automata Failures: A Calculus and a Method," IBM J. Res. Develop., vol.10, pp.278-291, July 1966.

(11) P.Goel,"An Implicit Enumeration Algorithm to Generate Tests for Combinational Logic Circuits", IEEE Trans. Comput., pp.215-222, Mar.1981.

(12) R.L.Wadsack, " Fault Modeling and Logic Simulation of CMOS and MOS Integrated Circuits," Bell System Technical Journal, vol.57, pp.1449-1473,May-June 1978.

(13) Y.M.El-Ziq and R.J.Cloutier,"Functional-Level Test Generation for Stuck-open Faults in CMOS VLSI," ITC'81,pp.536-546,1981.

(14) R.Chandramouli,"On Testing Stuck-open Faults," FTCS-13,pp.258-265, 1983.

(15) 梶原誠司，板崎徳禎，樹下行三："組合せ回路のスタックオープン故障のテストパターン生成について"，信学技法,FTS89-9,pp.15-20,June 1989.

(16) Y.Takamatu and K.Kinoshita, "CONT: A Concurrent Test Generation Algorithum," FTCS-16, pp.22-27,June 1986.

(17) 梶原誠司，板崎徳禎，樹下行三："可観測な環境を前提としたテストパターン生成",信学論D-I,vol.J73-D-I No.3,pp.342-349, Mar. 1990.

(18) P.R.Menon and S.G.Chappell,"Deductive Fault Simulation with Functional Blocks," IEEE Trans. Comput.,vol.C-27,pp.689-695,Aug 1978.

(19) E.G.Ulrich and T.Baher,"The Concurrent Simulation of Nearly Identical Digital Networks," 10th DAC,pp.145-150, 1973.

(20) Y.Miura and K.Kinoshita,"Testable Design for Stuck-Open Faults with the Robustness," Trans. IEICE,vol.E73,No.8,pp.1294-1300,Aug. 1990.

(21) M.K.Reddy and S.M.Reddy,"Detecting FET Stuck-open Faults in CMOS Latches and Flip-flops," IEEE Design and Test of Comput.,pp.17-27, Oct. 1986.

(22) A.Rubio,S.Kajihara and K.Kinoshita,"Fault Detection of Stuck-open Faults for Memory Elements,"信学技報，FTS90-13,pp.39-46,1990.

(23) F.Brglez,D.Bryan and K.Kozminski, "Combinational Profiles of Sequential Benchmark Circuits," ISCAS'89,pp1927-1934,June 1989.

(24) M.H.Schulz and E.Auth,"Advanced Automatic Test Pattern Generation and Redundancy Identification Techniques," FTCS-18,pp30-35,1988